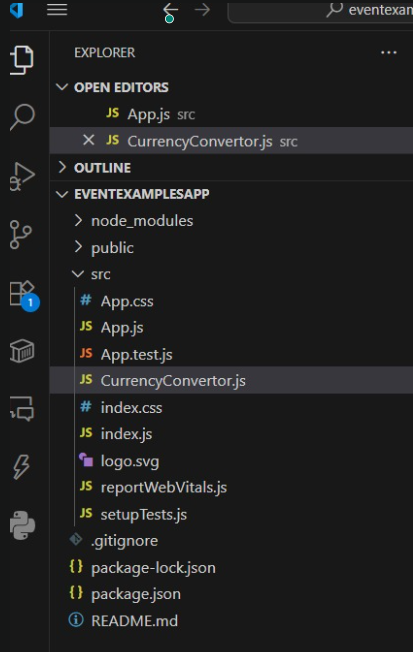
**REACT**

**React JS HOL (Handson 11)**

**Currency Convertor**

**FOLDER STRUCTURE:**

****

**PROGRAM:**

**App.js**

import React, { Component } from 'react';

import CurrencyConvertor from './CurrencyConvertor';

class App extends Component {

constructor() {

super();

this.state = {

count: 0

};

}

increment = () => {

this.setState({ count: this.state.count + 1 });

this.sayHello();

};

decrement = () => {

this.setState({ count: this.state.count - 1 });

};

sayHello = () => {

console.log("Hello! This is a static message.");

};

sayWelcome = (message) => {

alert(message);

};

handleSyntheticEvent = (event) => {

console.log("I was clicked");

};

render() {

return (

<div style={{ padding: '20px' }}>

<h1>React Event Handling Examples</h1>

<h2>Counter: {this.state.count}</h2>

<button onClick={this.increment}>Increment</button>

<button onClick={this.decrement} style={{ marginLeft: '10px' }}>Decrement</button>

<br /><br />

<button onClick={() => this.sayWelcome("Welcome to the event handler example!")}>

Say Welcome

</button>

<br /><br />

<button onClick={this.handleSyntheticEvent}>Click Me (Synthetic Event)</button>

<br /><br />

<CurrencyConvertor />

</div>

);

}

}

export default App;

**CurrencyConvertor.js**

import React, { useState } from 'react';

function CurrencyConvertor() {

const [rupees, setRupees] = useState('');

const [euros, setEuros] = useState(null);

const handleSubmit = (event) => {

event.preventDefault();

const converted = parseFloat(rupees) \* 0.011; // Conversion Rate: 1 INR ≈ 0.011 EUR

setEuros(converted.toFixed(2));

};

return (

<div>

<h2>Currency Converter</h2>

<form onSubmit={handleSubmit}>

<label>Enter Amount in ₹ (INR): </label>

<input

type="number"

value={rupees}

onChange={(e) => setRupees(e.target.value)}

/>

<button type="submit" style={{ marginLeft: '10px' }}>Convert</button>

</form>

{euros !== null && <p>Converted Amount: € {euros}</p>}

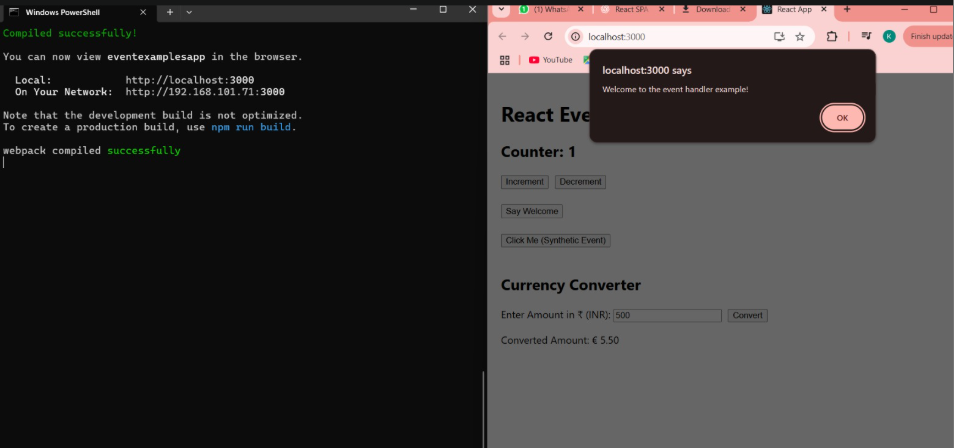
</div>

);

}

export default CurrencyConvertor;

**OUTPUT:**

****